

Friedrich-Schiller-Universität Jena
Institut für Geographie

Doreen Papendick, Ulrike Bende-Michl, Carsten Busch



Geohydrologische Datenbanken im Internet

- Abfrage, Datenmodellierung, Präsentation -

Studienprojekt im Bereich Geoinformatik

Gliederung



- ⌘ Nutzung Geohydrologischer Datenbestände
- ⌘ Datenbanksysteme und Internet
- ⌘ Vor- und Nachteile verschiedener Techniken
- ⌘ Vorstellung des Studienprojektes
 - ☒ Zielsetzung
 - ☒ Umsetzung und auftretende Probleme
 - ☒ Ergebnisse und Ausblick

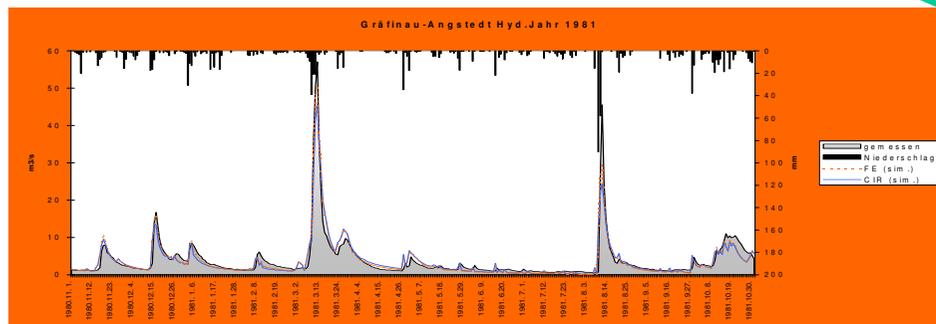
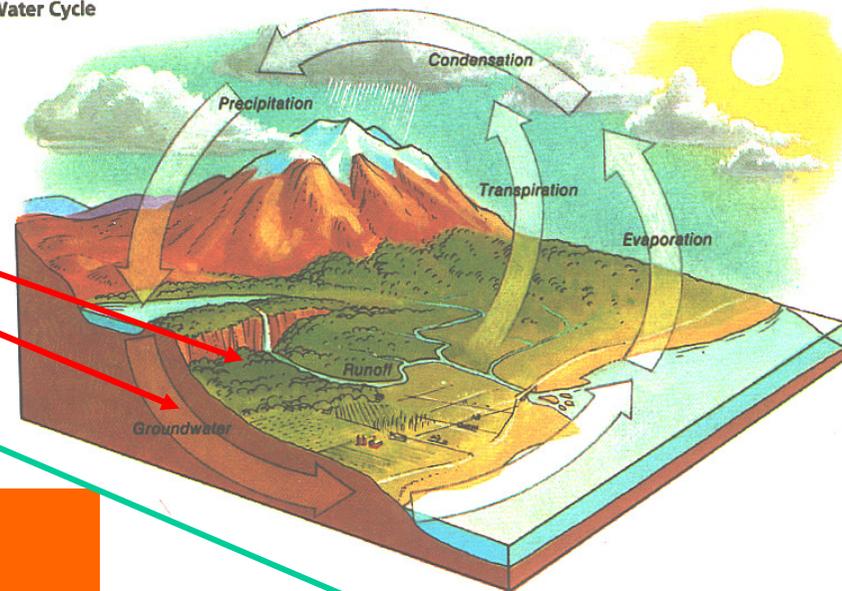
Geohydrologische Datenbestände

Bereitstellung von regionalen, umweltrelevanten Informationen zum Wasser- und Stoffkreislauf

Prozeßforschung

Modellierung & Simulation

The Water Cycle



Systemanalyse

Geohydrologische Datenbestände

Ergebnis:

⌘ Prozeßforschung und Modellierung

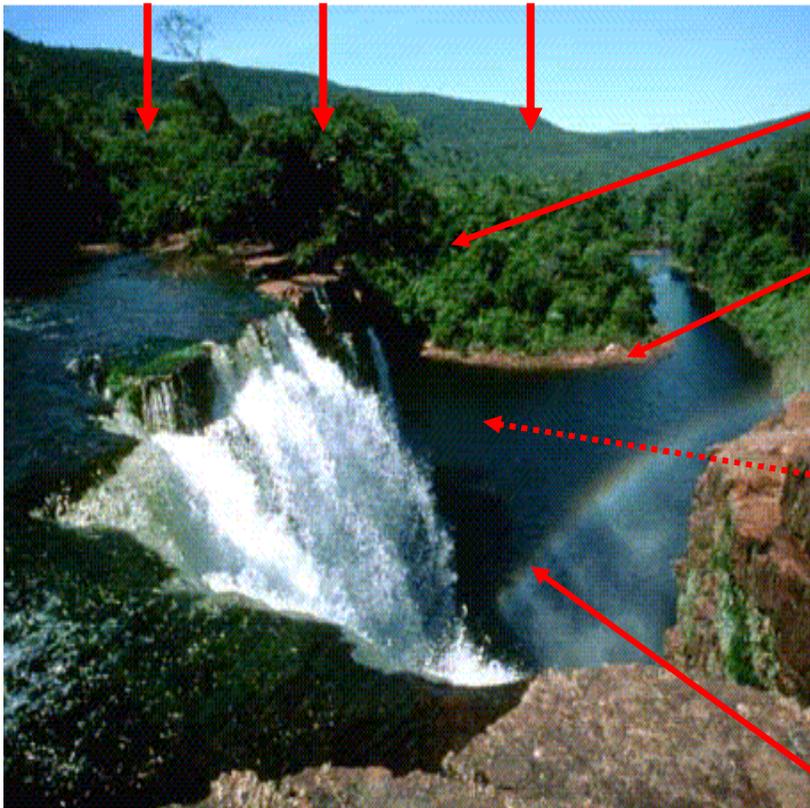
- Entstehung von Hochwässern
- Modifizierung hydrologischer Prozesse
- Veränderung der Wasserqualität

⌘ Wasserwirtschaftliche Entscheidungs- und Unterstützungssysteme

- Verfügbarkeit der Wasserquantität und -qualität
- What-if Szenarien

Geohydrologische Datenbestände

Erhebung prozessrelevanter Rauminformationen:



Landnutzungsinformationen: z.B.
Landnutzungsart, -management

Bodeninformationen: z.B. Art,
Körnung, organische
Gehalte

Flächeninformationen

Anthropogene Veränderungen: z.B.
Urbanisierung, Verschmutzungen,
etc.

Hydrometeorologische Informationen: z.B. Niederschlags-
Input, Temperatur

Zeitliche Informationen: z.B. Veränderung von
Wasservolumen, Wasserqualität

Punktinformationen

Geohydrologische Datenbestände

⌘ Problem:

- > Größe und Heterogenität der Dbestände
- > Kenntnis der DB-Struktur (z.B. ds_ids)
- > Kenntnis von SQL
- > Arbeiten nur auf dem DB-Server möglich

Geohydrologische Datenbestände

⌘ Motivation:

- Entwicklung eines Abfragewerkzeugs zur transparenten Ansicht auf die DB
- Interaktive Abfrage einzelner Parameter oder Parametergruppen
- Datenmodellierung für variable Zeiträume
- Interaktive deskriptive Datenstatistik

Geohydrologische Datenbestände

⌘ Beispieldatensätze

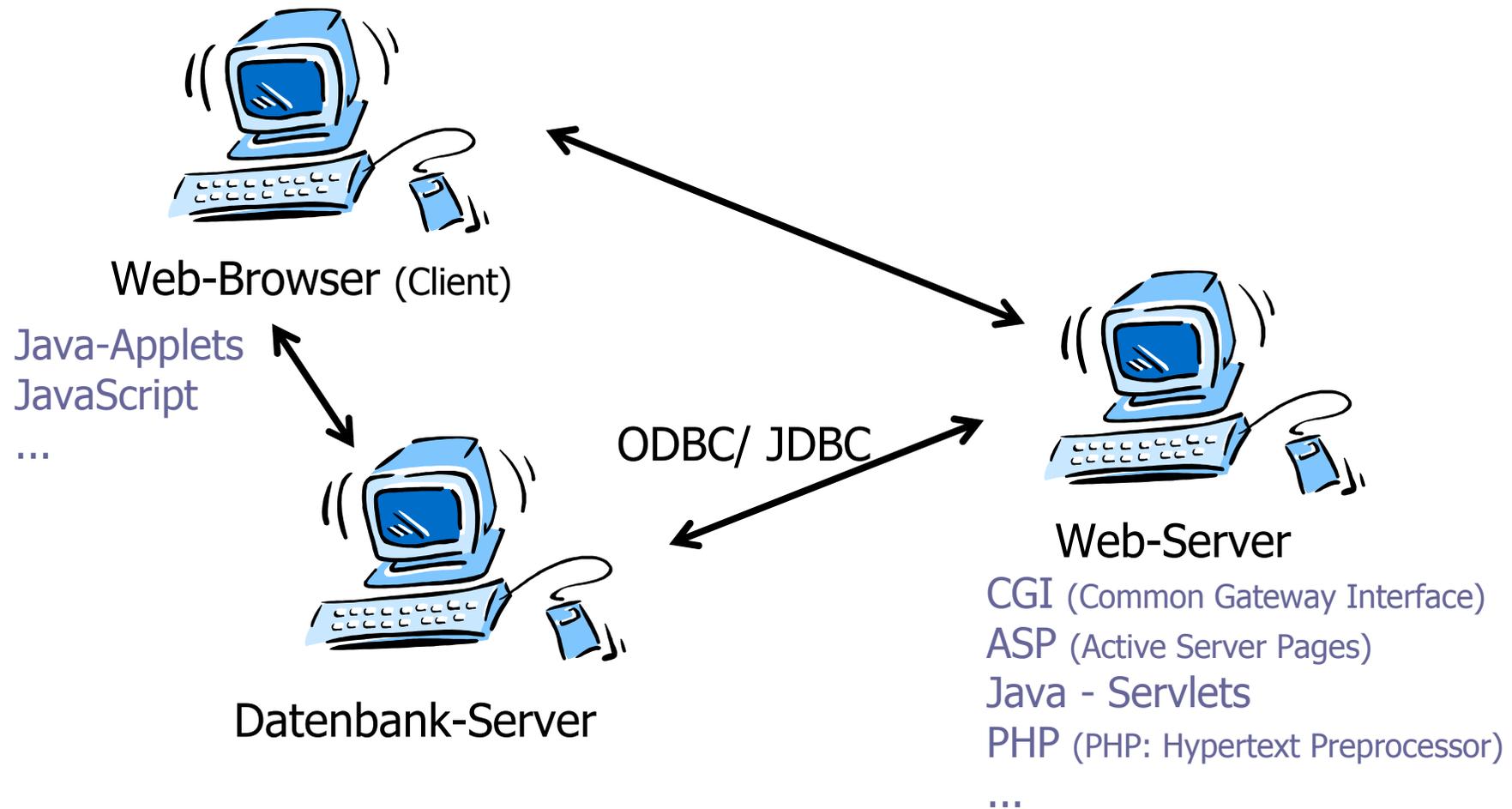


Datenbanksysteme und Internet

⌘ Anforderungen:

- ☒ leistungsfähige Schnittstellen
- ☒ hohe Flexibilität in Programmierung
- ☒ relativ einfache Integration von Datenbeständen einer Datenbank in Web - Seiten
- ☒ Session - Management
- ☒ hohe Performance
- ☒ Transaktionsmanagement
- ☒ keine Zusatzkenntnisse des Nutzers (besonders bezüglich Datenbankstruktur etc.)

Web - basierter Datenbankzugriff



Server-seitige Techniken (Übersicht)

nach Piessi, Ch. & E. Wilde, iX 3/2001, S. 88ff. [erweitert]

Anwendungen	Vorteile	Nachteile
CGI	hohe Flexibilität, verschiedene Programmiersprachen möglich (C/ C++, Perl, Tcl u.a.)	Start eines neuen Prozesses für jede HTTP-Anfrage, lange Startzeiten, Plattformabhängigkeit, Unterstützung von Sessions fehlen
Perl	leistungsfähige Programmiersprache, weit verbreitet	für komplizierte Webapplikationen sind zusätzliche Module notwendig, relativ langsam, da jeweils eigener Prozeß gestartet werden muß
PHP	leistungsfähige und einfache Schnittstellen zu Datenbanken u.a., Skripte in HTML integriert	eigene Sprache, Anbindung an bestehende Applikationen schwierig
ASP	Programmierung in JavaScript, VBScript sind bekannte und häufig verwendete Sprachen, Session-Unterstützung, Skripte in HTML integriert	Fähigkeiten von JavaScript sind begrenzt, nur im Microsoftbereich nutzbar
Servlet und JSP	Plattformunabhängigkeit, Objektorientierung, Prozeß nur einmal starten, Session-Management, parallele Bearbeitung mehrerer Anfragen	Web-Server benötigt Servlet-Unterstützung, Orientierung an API des jeweiligen Servers, Änderung in statischen Informationen (z.B. HTML-Code) führt zur Neuübersetzung des Servlets, bei JSP ist Java-Programmcode in HTML-Seiten integriert

Favoriten



⌘ Welche Methode?

☑ abhängig von:

- ☑ Aufgabenstellung und Zielsetzung
- ☑ Ressourcenverfügbarkeit
- ☑ Plattform
- ☑ Leistungsverhalten der Schnittstellen ...

Gliederung



- ⌘ Geohydrologische Datenbanken
- ⌘ Datenbanksysteme und Internet
- ⌘ Vor- und Nachteile verschiedener Techniken
- ⌘ Vorstellung des Studienprojektes
 - ☒ Zielsetzung
 - ☒ Umsetzung und auftretende Schwierigkeiten
 - ☒ Ergebnisse und Ausblick

Zielsetzung



- ⌘ Publikation von geohydrologischen Daten einer bereits bestehenden relationalen Datenbank im Internet
- ⌘ Welche Probleme bezüglich Datenbankstruktur, Schnittstellenimplementierung, Internetpräsentation und Anforderungen der Nutzer?

Besondere Anforderungen

- ⌘ Umrechnung von Maßeinheiten
- ⌘ Aufstellen von deskriptiven Statistiken
- ⌘ Berechnung und Darstellung unterschiedlicher zeitlicher Auflösungen
- ⌘ insbesondere Hydrologische Jahreswerte
 - ☒ Bsp.: 01.11.2000 - 31.10.2001 = hydrol. Jahr 2001

Realisierung der Datenabfrage

⌘ Java-Servlet

⌘ HTML-Seite mit speziellen Tags

- `<TEMPLATE> </TEMPLATE>`

⌘ und Variablen

- `&&NAME&&`

⌘ Servlet interpretiert HTML-Seite, fragt Daten aus Datenbank ab und generiert Ergebnisseite

Servlet - Aufruf



<!--Name von Server und Servlet-->

http://saale.geogr.uni-jena.de/servlet/DBXML
Map?

<!--Name des HTML-Templates-->

htmltemplatefile=projectresults.html&

<!--Parameterübergabe-->

Wildcard=&&PROJ&&"

Template - File

```
<!--<TEMPLATE SQL=SELECT ausdruck-1 [as alias-1]
      [,ausdruck-2 [as alias-2]]...
      FROM tabellenname;[BGCOLOR=("color-1","color-2");]>-->
<table>
  <tr BGCOLOR=&&BGCOLOR&&>
    <td>&&AUSDRUCK-1&& | &&ALIAS-1&&</td>
    [<td>&&AUSDRUCK-2&& | &&ALIAS-2&&</td>]...
  </tr>
<!--</TEMPLATE>-->
</table>
```

Zeitliche Auflösung



- ⌘ Zeitlicher Betrachtungsmaßstab
besonders bei Zeitreihen von Bedeutung
- ⌘ tägliche und monatliche Auflösung mit
ORACLE Aggregierungsfunktion *AVG*
(*average* - Mittelwert) realisiert
- ⌘ Problem: Mittelwertberechnung über
hydrologische Jahre

Hydrologische Jahre

ds_id	timestamp	value	hy_id
342	01.03.1999	23	1999
342	01.11.1999	14	2000

View

hy_id	startdate	enddate
1999	01.11.1998	31.10.1999
2000	01.11.1999	31.10.2000
...

Views - Management

⌘ Aufnahme von Transformationsberechnungen (Maßeinheiten)

⌘ Umrechnungen in Views (auf Basistabelle mit Originaleinheiten) definiert

→ externe Tabelle

basistable
viewname
abstract

Probleme



⌘ Gewährleistung einer nutzerorientierten Auswahl der Tabellenspalten

⌘ Realisierung mit Array

- ☒ nimmt Auswahl der Select-Box auf
- ☒ Trennung der Array-Positionen durch definierbare Trennzeichen
- ☒ wichtig für unterschiedliche Verwendung des Arrays

ARRAY



```
<!--<VAR arraydelimiter="</td><td>"/>-->
```

```
<td>&&COLUMNS&&</td>
```

```
<!--<VAR arraydelimiter=","/>-->
```

```
<!--<TEMPLATE SQL=SELECT timestamp,  
&&COLUMNS&& FROM tablename;;>-->
```

```
<!--<VAR arraydelimiter="&&&&</td><td>&&&&"/>-->
```

```
<td>&&&&COLUMNS&&&&</td>
```

Ergebnisse (I)

- ⌘ Einbettung von *SQL - Statements* und Ausgabevariablen in HTML - Quelltext
 - ☒ ohne Datenbankzugang ist HTML - Definition möglich
- ⌘ Weitergabe der Variablen durch Verweise oder Formulare
 - ☒ bei Zunahme der Parameter erfolgt Verlust der Übersichtlichkeit
 - ☒ visualisierte Aufstellung der Entscheidungs - bzw. Strukturpfade

Ergebnisse (II)



- ⌘ Abbildung einer schon existierenden Datenbankstruktur und Verwendung eines implementierten Servlets
 - ☒ auf Datenbank- und Servletebene waren geringe Änderungen notwendig
- ⌘ Die Datenbankstruktur konnte erfolgreich beschrieben und abgebildet werden.

Ausblick (I)



⌘ Visualisierung der geohydrologischen Messwerte

- ☑ Daten graphisch auszuwerten
 - ☒ Realisierung z.B. mit Java

⌘ räumliche Darstellung

- ☑ Erweiterung des relationalen Modells um geometrische Datentypen und Funktionen
 - ☒ Bsp.: GeoTask - Server
 - ☒ UDT (user definiert datatyps) und UDF (user definiert functions)

Ausblick (II)



⌘ Erweiterung um Statistikberechnungen

⌘ XML - basiertes Web - Publishing

☒ Bsp.: Apache - Projekt Cocoon

☒ Cocoon = Java - Servlet (Publishing Framework Servlet)

☒ Trennung von Inhalt, Geschäftslogik und Darstellung

☒ wandelt XML-Quelldokument je nach anfragendem Client (Browser) in ein beliebiges Zielformat um (z.B. HTML-Seite)

Fazit



- ⌘ trotz geringer Änderungen erfolgreiche Realisierung der Publizierung einer Datenbank im Internet mittels Servlet
- ⌘ projektübergreifende Anfragen der unterschiedlichen Arbeitsgruppen gegen die Datenbank vereinfacht
- ⌘ für den Nutzer keine Kenntnis der Struktur der Datenbank und der SQL - Abfragesprache erforderlich